

An Improved Network Server

Field of the Invention:

The present invention relates to communication systems and more particularly to networked servers.

Background:

Multi-processing, that is, simultaneously handling multiple tasks by periodically switching between a number of different tasks, is common in modern day computer systems and in network servers. There is a large body of literature relating to interrupt handling and to prioritize different tasks. However, in existing network servers the user has little if any control over the prioritization of the individual tasks which the server performs in responding to different types of specific requests that the server receives. Version 6 of the IP protocol does have a priority field; however, the purpose of this field is to prioritize the transmission of packets through the internet routers, rather than prioritizing tasks inside the server.

Summary of the present invention:

The present invention provides an event-driven system that provides scheduling and resource allocation for an internet server. The present invention utilizes a cost-benefit model and user preferences to prioritize and schedule tasks. A network server's performance is improved by prioritizing tasks according to their importance, cost, and the system owners desires. The tasks are scheduled and resources (for example memory) are allocated to the tasks in accordance with their priority. Interlayer communication is used to provide a faster way to move data and to provide feedback as to the current state of a particular layer. Header parsing and peeking provides a way to make decisions earlier rather than waiting for the necessary information to bubble up to a higher layer. A thin thread model is used to handle tasks. The progress of the thin threads relative to each other is monitored and controlled.

Brief Description of the Figures:

Figure 1 is a system block diagram showing the organization of a first embodiment of the present invention.

Figure 2 is a programming block diagram that illustrates the operation of the system.

1 Figure 3 illustrated an HTTP request which is processed by the present invention.

2
3 **Detailed Description:**

4 The first embodiment includes number of clients 100A to 100Z that are connected to a
5 server system 101 via the Internet. The clients 100A to 100Z may for example be personal
6 computers with may the Microsoft Windows operating system and a browser connected to
7 the internet via dial-up, DSL or cable modems. While only five such client systems 100A to
8 100Z are specifically shown in Figure 1, it should be understood that there may be
9 hundreds or even thousands of such client machines accessing a particular server 101
10 through the internet at a particular time. Only five are shown in Figure 1 for convenience of
11 illustration. The figures is intended to illustrate that there are a large number of client
12 machines sending requests to a server system 101.

13
14 The server system 101 includes a web application server 110 (which in fact may consist of
15 multiple servers) that respond to the requests from clients 100A to 100Z. The web
16 application server can be any of type of web server such as those that are accessible from
17 the internet. For example the web application server 110 can be a network interface to a to
18 a set of programs that generate web pages based on content from a variety of sources.
19 Often in such systems a database is involved. A specific example is the type of web server
20 typically used by brokerage companies. Such servers typically have web pages
21 constructed by running a program which uses information from a database which includes
22 account information together with pictures and logos, etc. A conventional net interface 122
23 receives requests for clients 100A to 100Z and passes them to other programs in the
24 system for further processing. The requests that come from clients 100 can for example be
25 a request such as the example shown in Figure 3. (Note the request is shown in a figure
26 since according to patent office rules, such material can not be included in the
27 specifications). The example shown in Figure 3 includes the command GET, the protocol
28 identification, in this case "http", a www server address, in this case "yahoo.com", followed
29 by the item name index.com. As is well known, various other information can be also be
30 included in such a request..

31
32 Each request received by the system is first classified by program 121, the request is then
33 prioritized by program 1122 and finally it is scheduled by program 124. A data base 123

specifies how different types of requests are to be classified. Data base 123 is set up and controlled by the system operator. The data 123 also specifies the cost of each type of request. The cost assigned to each type of request can be a fixed cost specified by the system operator for that type of request or it can be a dynamic type of cost function that takes into account the cost of processing the previous similar request. The classification of each request is done by program 121 according to the specifications and information set out in data base 123. The system owner can change the information in data base 123 from time to time as desired in order to make the system operate in a specific manner. The information in data base can be changed in a conventional manner such as via one of the clients 100A to 100Z which has system administrator privileges.

The classification could for example specify the benefit and cost of each request as follows: A request to order a product might be given a benefit rating of 20, whereas, a request to view a news article might be assigned a benefit rating of 10. Requests from the Company CEO for personnel information may be given a higher benefit rating than requests which come from the accounting department for historical accounting information.

Each request is also classified according to cost: A request to order a product might be assigned a cost of 50 whereas a request to view a news article might be assigned a cost of 20. The costs assigned to a request would take into account such factors as the amount of buffer memory required by the task, the amount of CPU time, the bandwidth requirements.

Given four different requests, they may be classified with cost and benefit parameters by

Request 1: benefit 20 cost 20

Request 2: benefit 20 cost 2

Request 3: benefit 10, cost 3

Request 3: benefit 20 cost 5.

After requests are classified, they are prioritized by program 122. Program 122 implements rules that the system operator stores in data base 123. That is, information in data base 123, specifies how requests are to be prioritized. The rules can be simple or complex. For example, The system owner might simply specify a rule that says: subtract the cost from the benefit and the resulting number is the priority. On the other hand the prioritization can